REQUIREMENT DEVELOPMENT PROCESS

Contents

[1. Process 3](#_Toc462954194)

[1.1. Process purpose 3](#_Toc462954195)

[1.2. Process activities / steps 4](#_Toc462954196)

[1.3. Activities description 4](#_Toc462954197)

[2. Role and responsibility 5](#_Toc462954198)

[3. Tools and methods 6](#_Toc462954199)

# Process

## Process purpose

Software Requirements is a field within software engineering that deals with establishing the needs of stakeholders that are to be solved by software. The IEEE Standard Glossary of Software Engineering Terminology defines a requirement as:

* A condition or capability needed by a user to solve a problem or achieve an objective.
* A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document.
* A documented representation of a condition or capability as in 1 or 2.

The activities related to working with software requirements can broadly be broken up into Elicitation, Analysis, Specification, and Management.

## Process activities / steps



## Activities description

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Objective | Input | Output | Activities |
| Elicitation | Understand business workflow, customer needs  , constraints of proposed system | Customer needs | Business workflow doc  Proposed system function doc with constraint and quality attribute | 1. Identify stakeholder 2. Meeting with customer 3. Gather customer requirement using: Req. E technique |
| Analysis | Analyze customer requirements to offer solutions for proposed system | Business workflow doc  Proposed system function doc with constraint and quality attribute | System prototype  Proposal doc | 1. Define scope of project 2. Identify customer problem 3. Use 4W+ 1H method 4. Review and give priority to requirements 5. Draw prototype 6. Make proposal doc |
| Specification | Specify the requirement into user and developer requirement to implement easier | System prototype  Proposal doc | ConOps doc  URD  SRS  Usecase doc  System prototype | 1. Write Requirement in structure for Customer, Developer and User to understand 2. Make document |
| Validation | Validate with stakeholder( customer) to make sure the requirement is correct | ConOps doc  URD  SRS  Usecase doc  System prototype | ConOps doc  URD  SRS  Usecase doc  System prototype | 1. Present document with customer 2. Re-evaluate 3. Re-write if needed |

# Role and responsibility

|  |  |  |
| --- | --- | --- |
| Role | Responsibility | Assign to |
| Requirement Leader | * Make and managephase plan * Set up meeting schedule with stakeholder * Prepare template to use in phase * Summarize and release phase document |  |
| Requirement Engineer | * Agree and apply method, technique to use in phase * Communicate, explore, gather Customer’s requirement * Define perspective of requirement, make prototype * Write document * Prioritize and validate requirement (complete, consistent) * Convert the user Req. to software Req. |  |
| Stakeholder (customer) | * Meet with team to provide and communicate requirement * Validate the document |  |
| Developer | * Involve in Specification to provide idea of Software function, help team understand better |  |
| Architect | * Involve in Specification to provide idea, help team understand better * Ensure the requirement match with scope * Identify the conflict of hardware, software, system |  |

# Tools and methods

|  |  |  |
| --- | --- | --- |
| Phase | Method used | Supported tool |
| Elicitation | * Interview * Questionaire * Brainstorm * Storyboarding * Analyze existing documents | End-user tool: AnnotatePro!, iRequire, ImmerdiateVisualization |
| Analysis | * Priortize * Use workflow, scenarios * Use dataflow diagram,state diagram * Prototyping(proposed system) * Conduct trade-off | Draw.io, ConTexter |
| Specification | * Prototyping * Use UML lang, flowcharts, swimlane * Use Use-cases | UML Language program |
| Validation | * Inspection * Review: walkthrough * Prototype | Checklist  Perspective-based reading  Creation of Artifacts |

# Goal, question, metric:

# Goal:

Goals are defined in term of purpose, perspective.

* Purpose: To analyze requirements to understand it and develop it.
* Perspective: Examines the requirement change from the point of view of thecustomer.

# Question:

The question for requirement development process:

* What data should be collected?
* What kind of the program that stakeholder want to develop?
* Is the data that give by customer clearly?
* How to collect data from customer?
* The data from the customer enough to build the software?

# Metrics:

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Formula | Purpose | Reference value |
| Unambiguous | Q =Nui / Nr | Percentage of requirement that have been interpreted by all reviewer | 0 = Ambiguous requirement  1 = Unambiguous requirement |
| Correct | Q = Nc / Nr | Percentage of all requirement that are valid | 0 = Incorrect  1 = Correct |
| Complete | Q = Nu / Ni \* Ns | The number of functions currently specified | Closer to one, the more complete |
| Understandable | Q = Nur / Nr | The number of requirements that are understood by all users and reviewers | 0 = No requirement understood  1 = All requirements understood |

* Nui: the number of requirement for which all reviewers presented identical interpretations
* Nr : total number of requirement
* Ni : the stimulates input of the function
* Ns: the stage input of the function
* Nur: the current unique functions requirement